Universidade Católica Dom Bosco  
Engenharia de Computação

Luan Henrique Santos Miranda   
João Henrique Schweitzer Rezende

Macro ImageJ

Campo Grande – MS

2024

Luan Henrique Santos Miranda  
João Henrique Schweitzer Rezende

Macro ImageJ

Projeto desenvolvido para apresentação na disciplina Visão Computacional do curso de Engenharia da Computação, Universidade Católica Dom Bosco.  
  
Professor: Hemerson Pistori

Campo Grande – MS

2024

**Relatório de Desenvolvimento – ImageJ**

**Introdução**

Este projeto visa desenvolver um software para análise de imagens, focado na detecção e quantificação de áreas verdes em folhas de plantas. O software utiliza técnicas de processamento de imagem para identificar áreas verdes, pintá-las de vermelho e calcular a área correspondente em pixels e centímetros quadrados. Esta ferramenta pode ser útil em estudos botânicos, agrícolas e ambientais, onde a análise precisa da saúde e crescimento das plantas é fundamental.

**Desenvolvimento**

O desenvolvimento do software foi realizado utilizando a linguagem de programação Python e a biblioteca OpenCV, que fornece um conjunto robusto de funções para processamento de imagens. O processo de desenvolvimento seguiu uma abordagem incremental, com testes e validações contínuas para garantir a precisão e eficiência das funções implementadas.  
*O Projeto foi desenvolvido com auxílio de inteligências artificiais como o Chat GPT 3.5.*

**Funcionalidades Principais**

Carregamento de Imagens: Carrega as imagens de folhas de plantas a partir do sistema de arquivos.

Detecção de Áreas Verdes: Identifica e destaca as áreas verdes da imagem utilizando a conversão de espaço de cores BGR para HSV e a aplicação de uma máscara de cor.

Pintura das Áreas Verdes: Pinta as áreas verdes detectadas de vermelho para facilitar a visualização e análise.

Cálculo da Área: Calcula a área das partes pintadas de vermelho em pixels e converte essa área para centímetros quadrados com base em uma proporção de pixels por centímetro.

Relatório de Resultados: Exibe os resultados do cálculo da área para comparações antes e depois do processamento.

#### **Componentes Principais do Código**

* Função carregar\_imagem: Carrega a imagem do arquivo especificado e trata erros de leitura.
* Função detectar\_verde: Converte a imagem para HSV e cria uma máscara para as regiões verdes.
* Função pintar\_verde\_com\_vermelho: Pinta as regiões verdes de vermelho usando a máscara.
* Função calcular\_area\_vermelha: Calcula a área das regiões vermelhas em pixels.
* Função calcular\_area\_vermelha\_cm2: Converte a área em pixels para centímetros quadrados.

**Cronograma de Desenvolvimento**

Planejamento: Definição de requisitos e escopo do projeto.

Implementação Inicial: Desenvolvimento das funções básicas de carregamento e detecção de verde.

Implementação de Funcionalidades Avançadas: Adição de cálculos de área e melhorias de performance.

Testes e Validação: Testes de precisão e correção de bugs.

Documentação e Relatório: Preparação da documentação e relatórios finais.

**Modo de uso**

Para utilizar o projeto, siga os seguintes passos abaixo:

Certifique-se de ter o “conda” instalado, após isso, dê o seguinte comando no terminal (base) para permitir que rode o script bash desenvolvido, para que então seja executada a aplicação de maneira correta:

chmod +x depimage.sh

Ainda no terminal o usuário deve dar o seguinte comando, iniciando assim, todo o processo de maneira automática, desde a criação do ambiente conda, instalação das dependências, até o processo de inicialização do código:

./depimage.sh

Uma vez que o arquivo bash foi executado e as bibliotecas foram corretamente instaladas, o código já irá mostrar o resultado da área de cada folha juntamente com a imagem delas.

**Conclusão**

O software desenvolvido proporciona uma ferramenta eficiente para a análise de áreas verdes em imagens de folhas de plantas. Utilizando técnicas de processamento de imagem, o software identifica e quantifica áreas verdes, facilitando estudos botânicos e agrícolas. A implementação modular e bem documentada permite futuras melhorias e adaptações conforme necessário. Este relatório documenta o processo de desenvolvimento, funcionalidades, e modo de uso, proporcionando uma visão abrangente do projeto e facilitando a replicação e utilização do software por outros interessados na área.

O tempo estimado para a finalização de todas as etapas do projeto é de **8** horas.

Link do vídeo do programa sendo executado: <https://youtu.be/MmgnbXd71OA>